

LTSpice

LTSpice is a free circuit simulator based on Berkeley's SPICE 3 program available for download from <http://www.linear.com>. This document describes some of the features used in EE105. Additional information is available from the tool's built-in help command and tutorials available on the web.

Common-Emitter Amplifier Example

Schematic diagram

To enter the schematic select the resistor and ground symbol from the tools menu and the transistor and voltage sources by clicking  and then choosing "npn" and "voltage" in the dialog that opens.

Set all device parameters (e.g. resistor value) by right-clicking on the component and entering the desired value(s) in the dialog box that opens.

Connect the devices with the wiring tool . Label nets with the  tool and drag components to make the schematic look pretty with . The larger hand, , also moves components but detaches them from their nets (wires).

To enter the .model and .dc statements, choose "Edit→SPICE Directive" from the menu and enter the text in the dialog box.

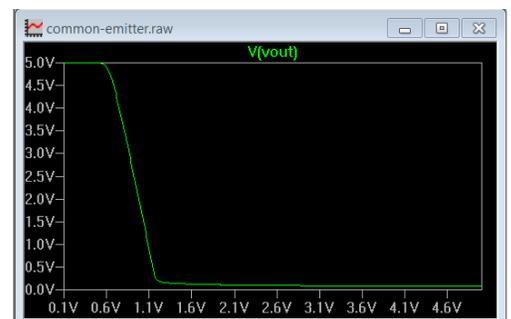
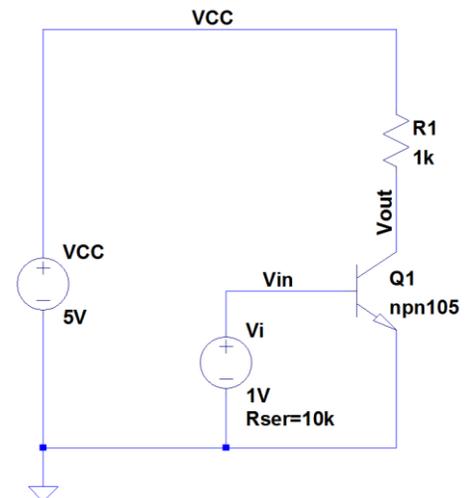
The .model statement defines a custom npn transistor with name npn105 and $I_s=10\text{fA}$, $\beta=100$, and Early voltage $V_A=100\text{V}$. Set the model parameter of Q1 to the name of the custom model (npn105), as shown in the example on the right.

Verify that your schematic looks just like the one shown above and that you have entered all component values. Rser is the series resistance of source V_i and limits the current flowing into the base of Q1. An alternative would be to reduce the maximum value of the DC sweep to less than 5V.

DC Analysis

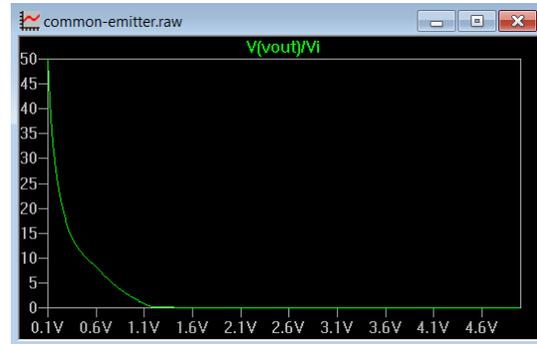
Clicking  runs the simulation. Since you have entered the analysis statement (.dc) in schematic, LTSpice knows that you want to perform a DC analysis. Otherwise a dialog box opens letting you choose the analysis to run. Plot V_{out} by clicking on its net or label in the schematic.

```
.model npn105 npn(is=10fA bf=100 vaf=100V)  
.dc Vi 0.1V 5V 0.01V
```

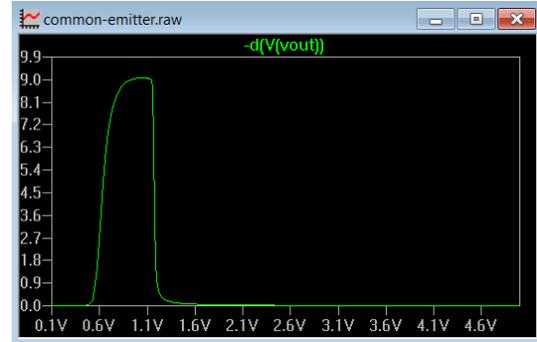


Plotting the Voltage Gain

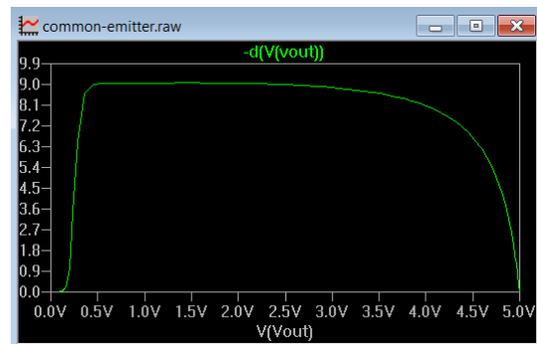
Use expressions to plot the voltage gain of the circuit. Right-click on “V(vout)” in the plot window and change the text in the dialog box to “V(vout)/Vi”. This plots the large-signal gain of the circuit shown on the right. Although we can compute and plot it, the large-signal gain of a common-emitter amplifier is not very meaningful.



To plot the small-signal gain $-a_v = -dV_{out}/dV_i$, change the plot expression to “-d(V(vout))”. As expected, the circuit has gain over a narrow range of input voltages.



Frequently it is more meaningful to plot a_v as a function of the output voltage V_{out} , rather than V_i . Click on the horizontal axis (e.g. on 2.6V) and enter “V(Vout)” for “Quantity Plotted”.



The plot changes to the one shown at the right, now indicating that that values on the horizontal axis are for V_{out} , rather than the analysis sweep variable, V_i . As we can see, the circuit has gain over a wide range of V_{out} . At low V_{out} , the gain drops as Q1 enters saturation. For large V_{out} , the bias current I_C drops (a result of the decreased voltage across R1), lowering the transconductance g_m of Q1.

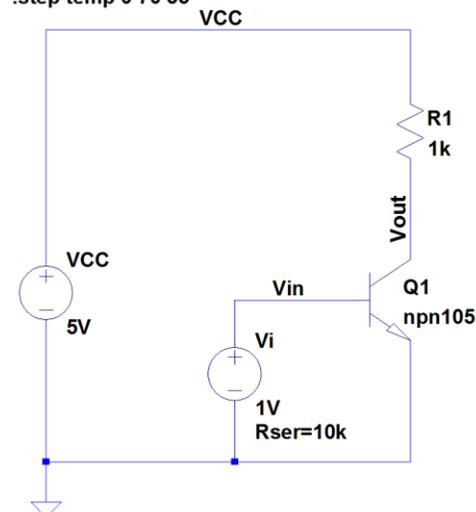
Simulating over Temperature

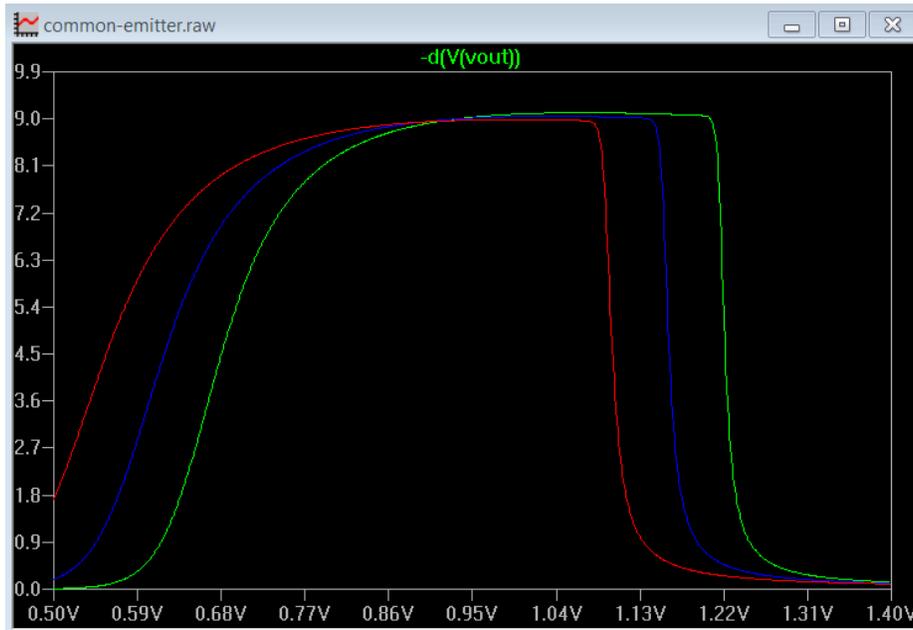
LTSpice has quite powerful features. Let’s simulate the gain variation as a function of temperature.

Change the .dc analysis statement as shown on the right and add the new SPICE directive “.step temp 0 70 35” to instruct SPICE to perform simulations between 0C and 70C in steps of 35C (i.e. to simulate at 0C, 35C, and 70C).

Rerun the simulation and configure the plot to show the small-signal gain as a function of V_i for the three different temperatures. Which trace corresponds to which temperature?

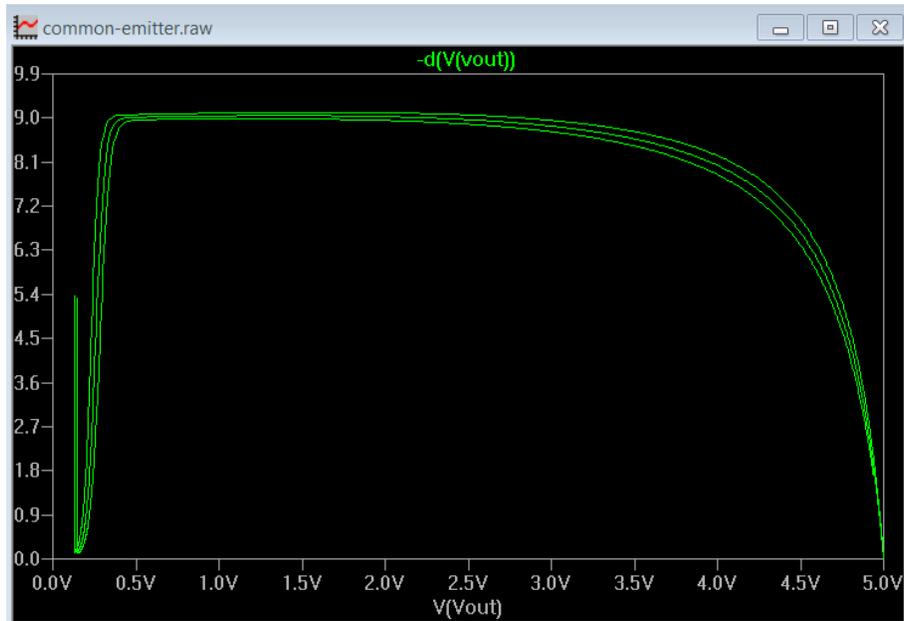
```
.model npn105 npn(is=10fA bf=100 vaf=100V)
.dc Vi 500mV 1.4V 1mV
.step temp 0 70 35
```





Small-signal gain versus V_{in} for temperatures 0C, 35C, and 70C.

Change the horizontal axis to V_{out} . Apparently the circuit gain only weakly depends on temperature. However, the bias point, i.e. the value of V_{in} for which the circuit has high gain, changes as a function of temperature. This is to be expected as $V_{BE(on)}$ decreases -2mV/C , shifting the $-a_v$ plot to the left as the temperature increases.



Small-signal gain versus V_{out} for temperatures 0C, 35C, and 70C.

Transfer Function Analysis

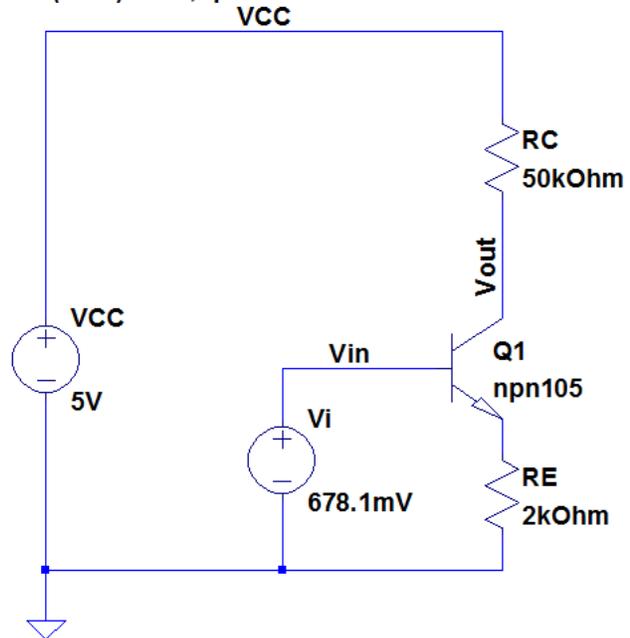
The SPICE transfer function analysis calculates the low frequency gain and input & output resistances of a circuit. We demonstrate the command for the CE amplifier with degeneration shown on the right.

If the circuit is biased such that $V_O = V_{CC}/2 = 2.5V$, we get $I_C = 50\mu A$, and $R_{in} \approx 252k\Omega$, $R_{out} \approx 50k\Omega$, and $a_v \approx -20$.

To simulate the circuit, first run a DC sweep to determine the bias value of V_i for with $V_O = 2.5V$. For circuits with high gain, you need to determine this value to many digits accuracy. Next run an operating point analysis (“DC op pnt” in LTSpice) and verify that the circuit is correctly biased. In this case check that $I_C = 50\mu A$ and $V_O = 2.5V$. This check is often an enormous time saver, as incorrect bias can lead to strange results that can take hours to debug. You have been warned!

Finally run a DC transfer function analysis. You need to specify the source (V_i in this case) and output of the circuit, $V(vout)$.

```
.model npn105 npn(is=10fA bf=100 vaf=100V)
;dc Vi 676mV 680mV 10uV
.tf V(vout) Vi ;op
```



```
* C:\Program Files (x86)\LTC\LTspiceIV\CE-deg.asc
--- Operating Point ---
V(vout) :      2.50097      voltage
V(vin) :       0.6781      voltage
V(n001) :      0.100943    voltage
V(vcc) :       5          voltage
Ic(Q1) :      4.99806e-005  device_current
Ib(Q1) :      4.90857e-007  device_current
Ie(Q1) :     -5.04715e-005  device_current
I(Rc) :       4.99806e-005  device_current
I(Re) :       5.04715e-005  device_current
I(Vcc) :     -4.99806e-005  device_current
I(Vi) :     -4.90857e-007  device_current
```

Output from the operating point analysis. Always perform this simulation to check the bias point of the circuit. In this case, $I_C = 50\mu A$ and $V_O = 2.5V$, as desired.

```

LT * C:\Program Files (x86)\LTC\LTspiceIV\CE-deg.asc
--- Transfer Function ---
Transfer_function:          -19.6002      transfer
vi#Input_impedance:        253217       impedance
output_impedance_at_V(vout): 49741.5    impedance

```

Output from the DC transfer function analysis. The simulated gain, -19.6, is close to the calculated value -20 which ignored the effect of r_o and made other simplifications. Same for the input and output resistance of the circuit.

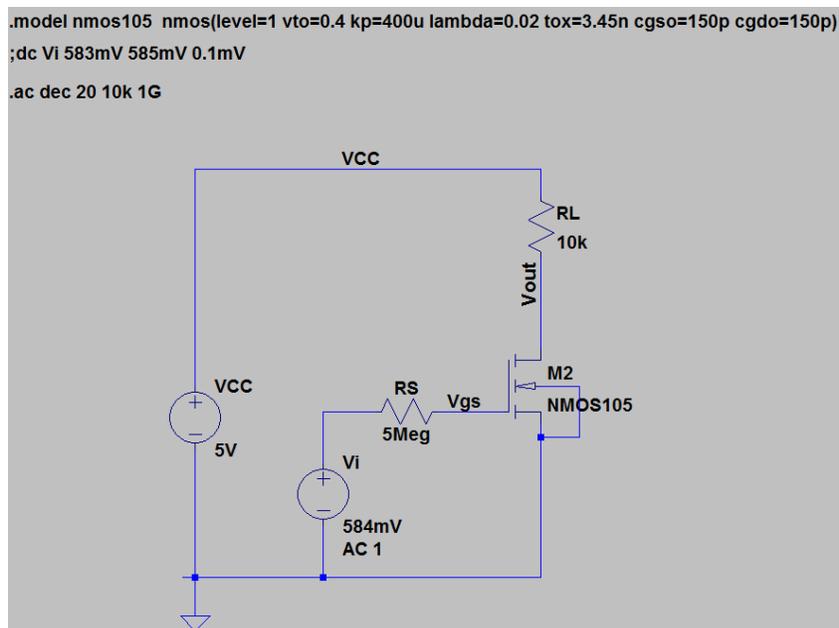
AC Analysis

The figure shows an example for an ac analysis. Determine the parameters in the .model statement

from the transistor specifications. Definitions: $k_p = \mu C_{ox}$, $t_{ox} = \frac{3.9\epsilon_o}{C_{ox}}$ ($\epsilon_o = 8.854 \frac{pF}{m}$) and

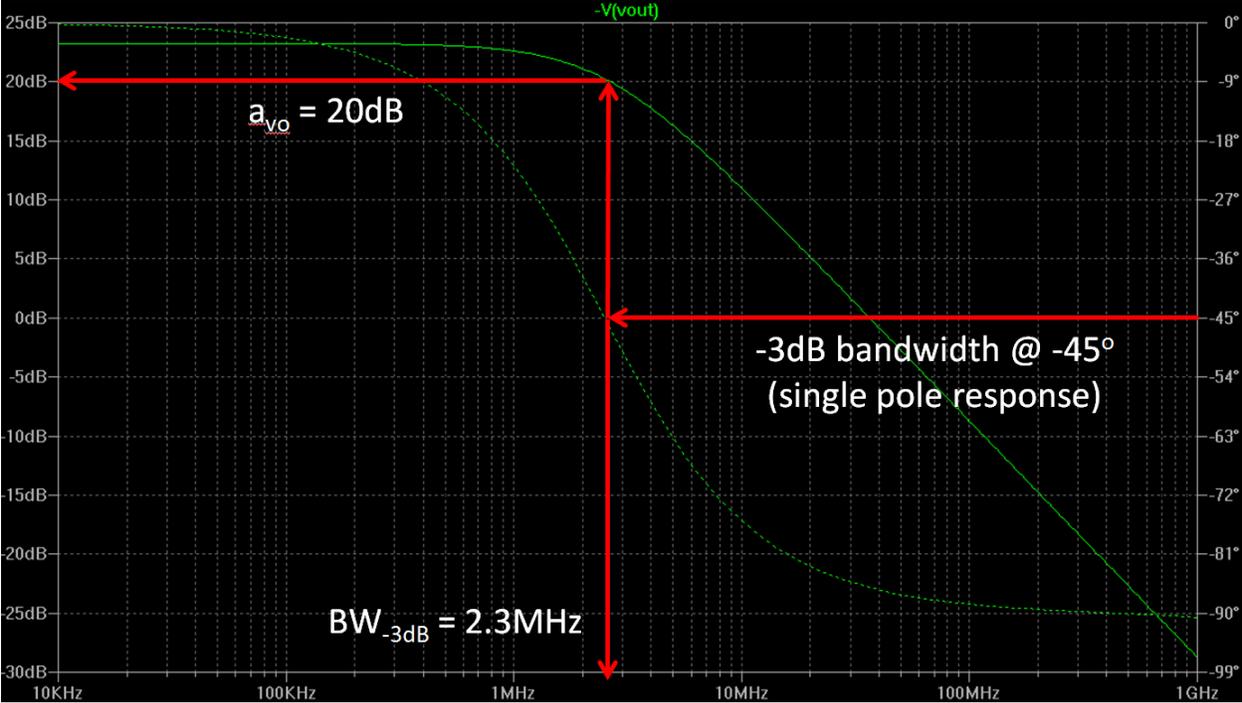
$C_{gso} = C_{gdo} = C_{ol}$. Use an LTSpice nmos4 device for the transistor and specify the model name, length, and width in the parameter dialog. Note: Spice will report an error that W and L are smaller than recommended for level 1 models. This warning refers to the fact that the simple “square law models” used in class are inaccurate for transistors with channel length less than about $2\mu m$. In actual design you would have to use more accurate (and complicated) models.

Next you need to specify an input. Use an LTSpice voltage component and determine and enter the bias voltage by running a DC analysis. Click on the “advanced” tab in the parameter section and enter AC Amplitude 1.



Sample schematic for ac-analysis.

The figure below shows the output (magnitude and phase) from the ac analysis. Always annotate SPICE output.



SPICE ac-analysis output.